Person-Target Profiling for Selection & Recruitment

CD Analysis

Part 2: The Coefficient Distribution Analyzer

Paul Barrett

Director of Research and Development

Psytech International (UK) Ltd.

paul.barrett@psytech.co.nz

Adjunct Professor of Psychometrics
& Performance Measurement
University of Auckland
Faculty of Business

paul.barrett@auckland.ac.nz

Adjunct Associate Professor of Psychology

University of Canterbury
Department of Psychology

paul.barrett@canterbury.ac.nz

Web: www.pbarrett.net

The Purpose

❶ To determine the sensitivity of 9 profile similarity/ distance coefficients to differences in elevation (level) and scatter (variability) whilst preserving an almost identical profile shape (*transformation sensitivity*)

Given a profile (1..*n*) of scores (*x*)...

$$Elevation = \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$Scatter = \left(\sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}} \right) \cdot \sqrt{n}$$

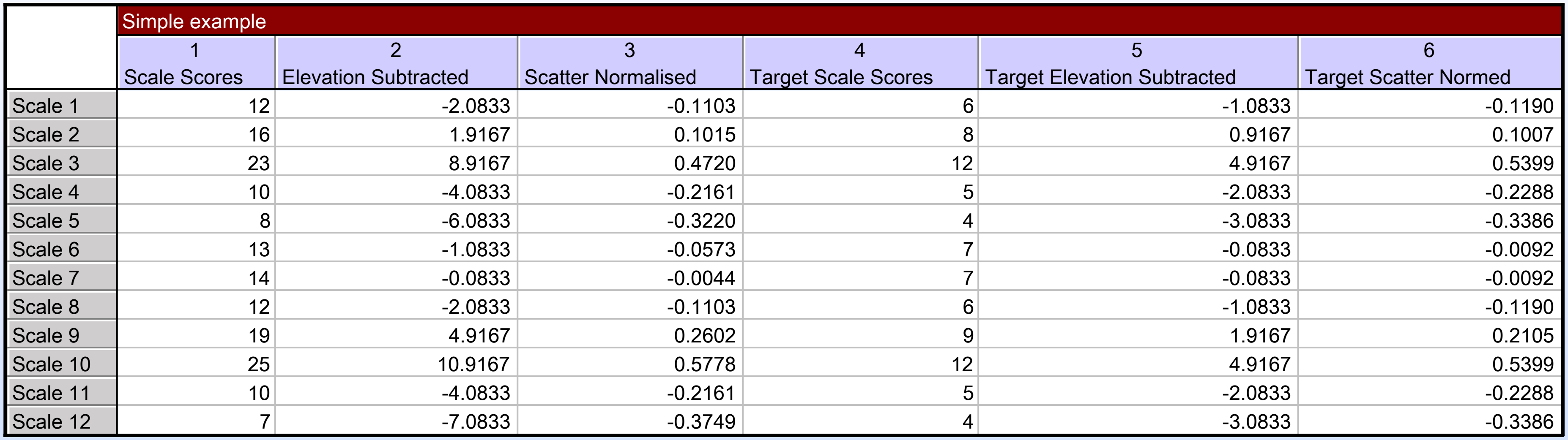
The Purpose

❷ To determine the expected size and frequency distribution of all possible values for each of the 9 profile coefficients, given a completely random dataset and a real-world personality test dataset (*distribution analysis*)

❸ To determine the sensitivity of certain coefficients to carefully graduated disparity between two profiles (*profile degredation analysis*).

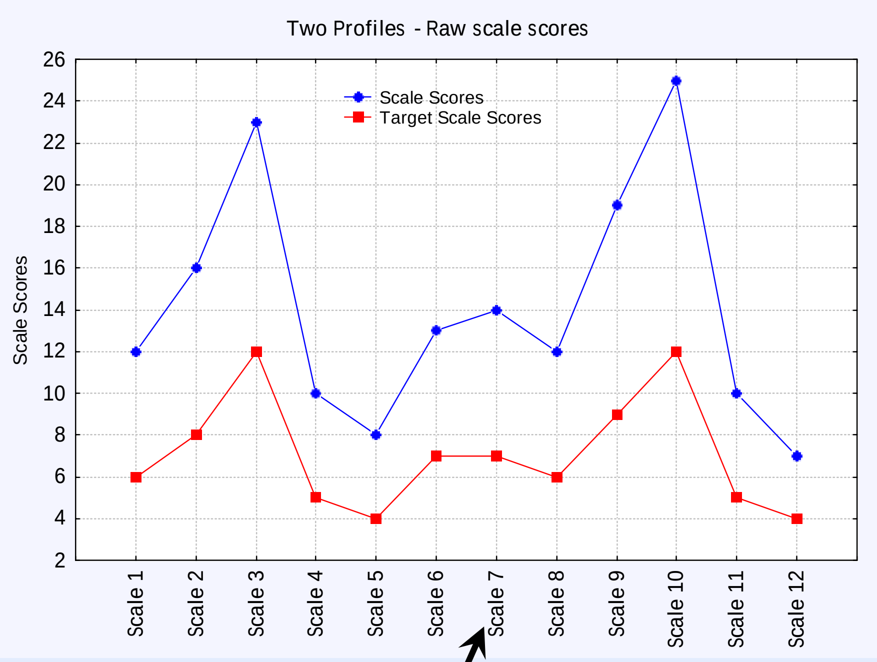
Transformation Sensitivity

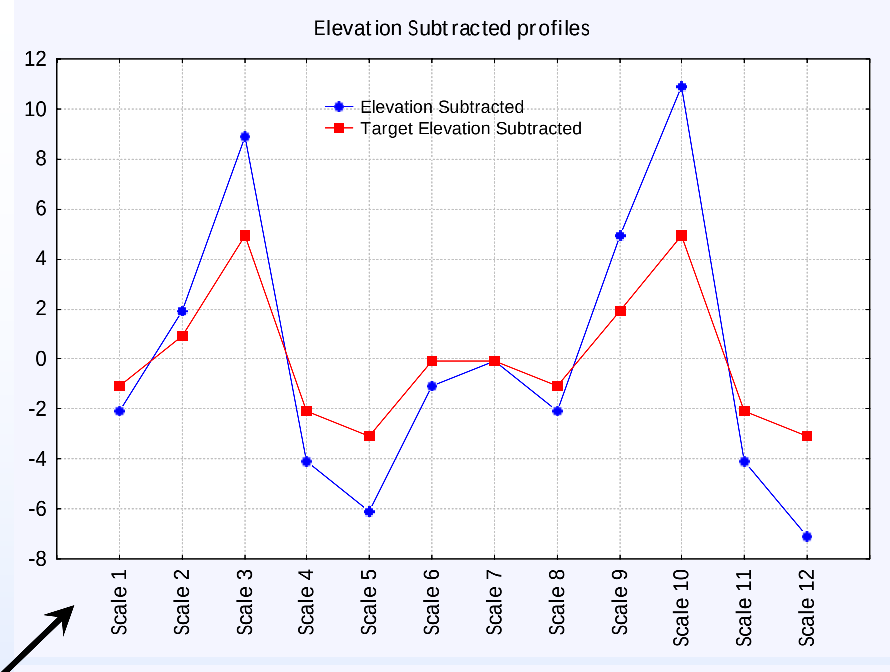
The dataset:

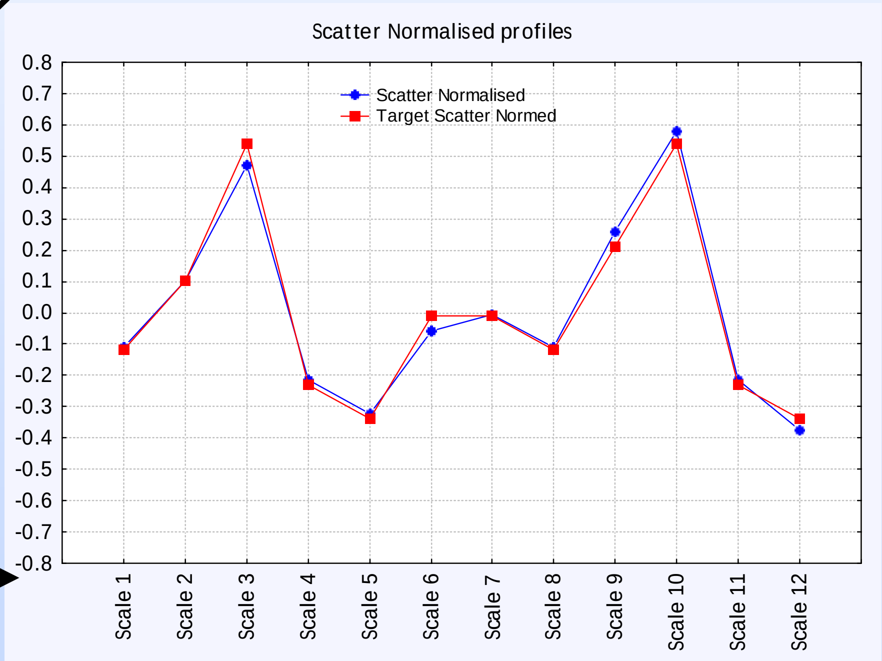


	Simple example					
	1 Scale Scores	2 Elevation Subtracted	3 Scatter Normalised	4 Target Scale Scores	5 Target Elevation Subtracted	6 Target Scatter Normed
Scale 1	12	-2.0833	-0.1103	6	-1.0833	-0.1190
Scale 2	16	1.9167	0.1015	8	0.9167	0.1007
Scale 3	23	8.9167	0.4720	12	4.9167	0.5399
Scale 4	10	-4.0833	-0.2161	5	-2.0833	-0.2288
Scale 5	8	-6.0833	-0.3220	4	-3.0833	-0.3386
Scale 6	13	-1.0833	-0.0573	7	-0.0833	-0.0092
Scale 7	14	-0.0833	-0.0044	7	-0.0833	-0.0092
Scale 8	12	-2.0833	-0.1103	6	-1.0833	-0.1190
Scale 9	19	4.9167	0.2602	9	1.9167	0.2105
Scale 10	25	10.9167	0.5778	12	4.9167	0.5399
Scale 11	10	-4.0833	-0.2161	5	-2.0833	-0.2288
Scale 12	7	-7.0833	-0.3749	4	-3.0833	-0.3386

Transformation Sensitivity







Raw Scores

Elevation (mean) subtracted

Scatter subtracted

Transformation Sensitivity: covariance-based coefficients

Pearson correlation

$r = \dfrac{\mathrm{cov}_{xy}}{s_x \cdot s_y}$ where N=the number of paired observations

cov_{xy} is the covariance between variables x and y calculated as:

$$\mathrm{cov}_{xy} = \frac{\sum_{i=1}^{N}(x_i - \overline{x})\cdot(y_i - \overline{y})}{N-1} \equiv \frac{\sum_{i=1}^{N} x_i y_i - \left(\dfrac{\sum_{i=1}^{N} x_i \sum_{i=i}^{N} y_i}{N}\right)}{N-1}$$

$$s_x = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}} \equiv \sqrt{\frac{\sum_{i=1}^{N} x^2 - N\cdot\overline{x}^2}{N-1}}$$

and likewise for s_y

Transformation Sensitivity: covariance-based coefficients

Pearson correlation

The coefficient varies between +1.0 and -1.0, with +1.0 indicating maximum similarity, -1.0 = maximum inverse similarity, and 0.0 indicating no relationship whatsoever.

The coefficient is sensitive only to shape, as both elevation and scatter are equated by the deviation and normalisation by standard deviation.

	Pearson r
Raw profile comparison	0.99
De-Elevation Comparison	0.99
De-Scatter Comparison	0.99

Transformation Sensitivity: covariance-based coefficients

Intraclass correlation

Model 2: We assume the profiles are randomly selected **(sampled)** from some population of profiles (a two-way random effects model). That is, the target profile is considered an instance from some population of all possible profiles, as is the profile from an individual (which is to be compared to the target). This means that we consider that we have sampled an individual's profile from some population of possible profiles. Likewise the target profile.

$$r_{icc} = \frac{MS_p - MS_{res}}{MS_p + (n_r - 1) * MS_{res} + \left(\frac{n_r * (MS_r - MS_{res})}{n_e} \right)}$$

where MS_p = Between profiles mean square

MS_{res} = Residual (interaction) mean square

MS_r = Between profile elements (scale scores) mean square

n_r = The number of profiles (=2)

n_e = The number of elements/scales in a profile

Transformation Sensitivity: covariance-based coefficients

Intraclass correlation

Model 2: may also be expressed algebraically as below …

$$r_{icc} = \frac{s_i^2 - s_d^2}{\left[s_i^2 + s_d^2 + \left(\frac{2}{n_e} \right) \cdot \left(n_e \cdot \bar{d}^{\,2} - s_d^2 \right) \right]}$$

where

$p =$ a profile element for the comparison profile

$t =$ a profile element for the target profile

$i = p + t$ and $d = p - t$

$s_i^2 =$ the variance of the i composite additive variable

$s_d^2 =$ the variance of the d composite difference variable

$\bar{d} =$ the mean difference score

$n_e =$ the number of elements in the profiles being compared

Transformation Sensitivity: covariance-based coefficients

Intraclass correlation

Model 3: We assume the profiles are the "population" profiles for a target and individual to be compared to that target profile (a two-way, fixed-effects model). It is assumed that these are the only two profiles - they are not "samples" from some theoretical population of possible target and individual profiles (in that the target is unique, and the individual profile is considered unique).

$$r_{icc} = \frac{MS_p - MS_{res}}{MS_p + (n_r - 1) * MS_{res}}$$

where MS_p = Between profiles mean square

MS_{res} = Residual (interaction) mean square

n_r = The number of profiles (=2)

Transformation Sensitivity: covariance-based coefficients

Intraclass correlation

The coefficient varies between +1.0 and -1.0, with +1.0 indicating maximum similarity, -1.0 = maximum inverse similarity, and 0.0 indicating no relationship whatsoever. Note that in many books, the "valid" range of an ICC is from 0 to +1.0 only.

The coefficient is differentially sensitive to elevation, scatter, and shape, depending upon the model chosen

	Model 2 ICC	Model 3 ICC
Raw profile comparison	0.35	0.78
De-Elevation Comparison	0.79	0.78
De-Scatter Comparison	0.99	0.99

Transformation Sensitivity: covariance-based coefficients

Congruence, Guttman's μ, and Alienation

$$r_c = \frac{\sum_{i=1}^{N} p_i \cdot t_i}{\sqrt{\sum_{i=1}^{N} p_i^2} \cdot \sqrt{\sum_{i=1}^{N} t_i^2}} \equiv \text{Guttman's } \mu \text{ ..... where}$$

$p =$ the comparison profile scores and

$t =$ the target profile scores (for $i = 1..N$)

Note that if p and t are expressed as deviation scores around their respective means

$$r_c = \frac{\sum_{i=1}^{N} (p_i - \overline{p}) \cdot (t_i - \overline{t})}{\sqrt{\sum_{i=1}^{N} (p_i - \overline{p})^2} \cdot \sqrt{\sum_{i=1}^{N} (t_i - \overline{t})^2}} \equiv \frac{\text{cov}_{xy}}{s_x \cdot s_y} \equiv \text{Pearson r}$$

$$Alienation(K) = \sqrt{(1 - \mu^2)} \equiv \sqrt{(1 - r_c^2)}$$

Transformation Sensitivity: covariance-based coefficients

Congruence, Guttman's μ, and Alienation

The congruence coefficient varies between +1.0 and -1.0, with +1.0 indicating maximum similarity, -1.0 = maximum inverse similarity, and 0.0 indicating no relationship whatsoever. Alienation varies between 0.0 and +1.0, with 0.0 = minimum alienation. (1-Alienation = a measure of "similarity").

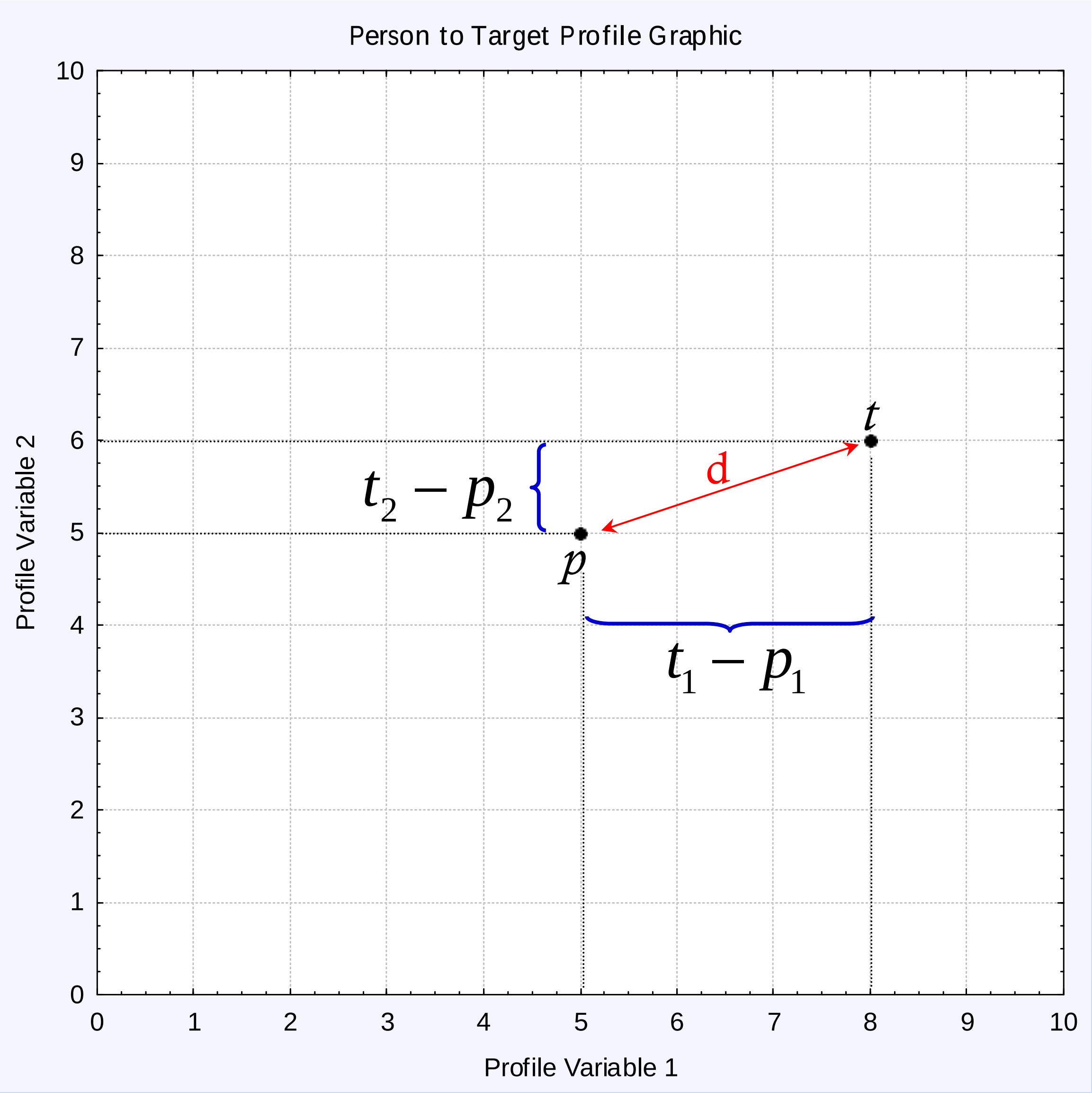
Both coefficients are sensitive only to shape.

	Congruence	**(1-Alienation)**
Raw profile comparison	1.0	0.96
De-Elevation Comparison	0.99	0.89
De-Scatter Comparison	0.99	0.89

Transformation Sensitivity: distance-based coefficients

Euclidean Distance

The Euclidean metric is that which corresponds to everyday experience and perceptions i.e the kind of 1, 2, and 3-Dimensional linear metric world where the distance between any two points in space corresponds to the length of a straight line drawn between them.



Transformation Sensitivity: distance-based coefficients

Euclidean Distance

$$d = \sqrt{\sum_{i=1}^{N} (t_i - p_i)^2}$$

where

t_i = the target profile score on scale i of N scales

p_i = the person profile score on scale i of N scales

Also known as Cronbach and Gleser's D, and Osgood and Suci's d

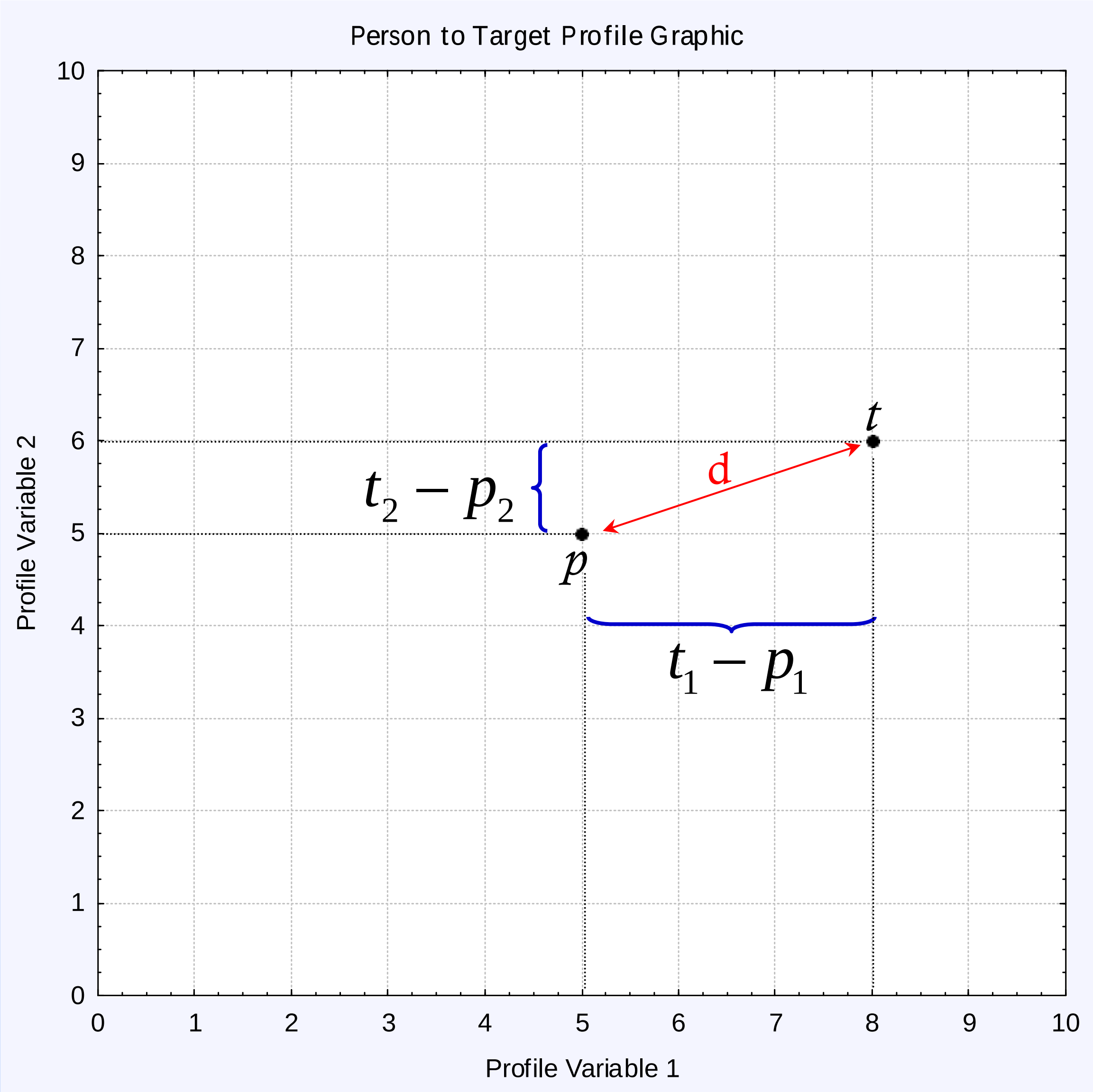
Transformation Sensitivity: distance-based coefficients

Euclidean Distance

$$d = \sqrt{\left(t_1 - p_1\right)^2 + \left(t_2 - p_2\right)^2}$$

$$d = \sqrt{\left(8-5\right)^2 + \left(6-5\right)^2}$$

$$d = \sqrt{9+1=3.162}$$



Transformation Sensitivity: distance-based coefficients

Euclidean Distance

Raw euclidean distance has no obvious bound value for the maximum distance, merely one that says 0 = absolute identity. Its range of values vary from 0 (absolute identity) to some maximum possible discrepancy value which remains unknown until specifically computed. So. profiles can be compared amongst one another in terms of their euclidean distance, given they are expressed in the same metric range as one another - but the distances permit only the relative ordering of profiles amongst one another, without regard to what the distance values imply in terms of absolute disparity. Put simply, euclidean distance varies as a function of the size of the observations.

Transformation Sensitivity: distance-based coefficients

Normalised Euclidean Distance d_n

In our example, the distance was 3.162. If all we saw were this number, what would we conclude? That the difference is “trivial”? What if the maximum possible distance that we could ever observe, given the range of scale scores, was 3.2? This would inform us that the two profiles are maximally discrepant. Conversely, if the maximum possible distance we might observe was 100, then 3.61 looks like a very good match. Thus, all such distance-based measures, are really only useful when expressed as **normalised** values, where they are normalised by the maximum possible distance that can be calculated using the range of the variables entering the calculations.

Transformation Sensitivity: distance-based coefficients

Normalised Euclidean Distance d_n

Thus, in our example above, the scores for the two variables ranged between 0 and 10. We can compute the maximum distance possible by assuming target 1 scores are {0, 0} and person scores are (10, 10). This is the maximum possible score discrepancy between a target and profile score with these variables. This discrepancy yields a raw maximum euclidean distance of 14.1421.

$$d = \sqrt{\left(t_1 - p_1\right)^2 + \left(t_2 - p_2\right)^2}$$

$$d = \sqrt{\left(0 - 10\right)^2 + \left(0 - 10\right)^2}$$

$$d = \sqrt{100 + 100 = 14.1421}$$

Transformation Sensitivity: distance-based coefficients

Normalised Euclidean Distance d_n

This is the maximum possible discrepancy – the normalisation factor. We can thus normalise all euclidean distances using these two profile variables into a 0.0 to +1.0 range, where 0 = absolute identity and +1.0 = maximum discrepancy.

In our example, the actual distance between our target and person profile was 3.162. Normalising this we obtain a normalised euclidean distance of 3.162/14.1421 = 0.22.

If we wish to express this as a “similarity” measure, we simply subtract it from 1.0 … = 0.78. We can also multiply this number by 100 in order to express it as % similarity.

Transformation Sensitivity: distance-based coefficients

Normalised Euclidean Distance d_n

The coefficient varies between 0.0 and +1.0, with +1.0 indicating maximum discrepancy, and 0.0 indicating absolute identity. Also can be expressed as % similarity using the formula: $s_d = \left(1 - d_n\right) \cdot 100.0$

The coefficient is differentially sensitive to elevation, scatter, and shape.

	Raw Euclidean	Normed Euclidean	% similarity
Raw profile comparison	26.19	0.348	65.2%
De-Elevation Comparison	9.90	0.188	81.19%
De-Scatter Comparison	0.114	0.057	94.30%

Transformation Sensitivity: distance-based coefficients

Mean Absolute Deviation

This is a very simple distance coefficient, sometimes refereed to as MAD - that uses simple absolute discrepancy as the measure of distance, and takes the average of these as the final coefficient value. Of course, as with *d*, the coefficient is scale-dependent (its values are expressed relative to the size of numbers being compared), so the most useful measure is the **normalised MAD** (*nMAD*). Also can be expressed as a similarity index using (*1-nMAD*).

Transformation Sensitivity: distance-based coefficients

Mean Absolute Deviation

The coefficient varies between 0.0 and +1.0, with +1.0 indicating maximum discrepancy, and 0.0 indicating absolute identity. Because it takes the absolute value deviation, it is insensitive to the direction of any differences and only sensitive to their magnitude (as is euclidean distance *d*)

The coefficient is differentially sensitive to elevation, scatter, and shape.

	1-nMAD
Raw profile comparison	0.77
De-Elevation Comparison	0.84
De-Scatter Comparison	0.99

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

The profile similarity coefficient (Cattell, 1969), ***rp***, or what has also been called the ***pattern similarity coefficient*** (Cattell, Coulter, and Tsuijoka, 1966; Cattell, Eber, and Tatsuoka, 1970; Cattell, 1978) was first introduced by Cattell in 1949.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

It was designed by Cattell (*taken from: Cattell, Coulter, and Tsuijoka, 1966, p.296*) to:

❶ take into account the metric and number of dimensions comprising the profiles to be compared.
❷ compare the coefficient with the magnitude to be expected by chance.
❸ provide a convenient function which behaves e.g. as regards distribution, in essentially the same general way as a Pearson r, varying from +1.0 indicating complete agreement between profiles to 0 for no relation, and -1.0 for complete inverse relation.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

The meaning of an ***rp*** of +1.0 is that two persons or patterns have exactly the same profiles and fall on the same point in multi-dimensional space. A value of 0 indicates that they fall as far apart as would be expected for any two points taken at random. A value of -1.0 means that they are at opposite ends of the distribution. Since the ends of a distribution are ill-defined and asymptotic, the value -1.0 is in actual practice approached but never quite reached, and there is in consequence a small asymmetry (positive skewing) in the distribution of rp about its median value of 0.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

Importantly, this coefficient provides a significance test for its value - the null hypothesis being that a coefficient observed of size *X* has been sampled from a population distribution where the mean value is 0. Put another way, the comparison profile is more similar to the target profile than expected by chance alone. Horn (1961) provides the formula derivation and tables for such a test - being based as it is on a chi-square variate. However, such a significance test is about as useful as one for an interrater reliability coefficient - none whatsoever.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

There are many different formulae for an r_p depending upon the kind of comparison being made:

- two individual person profiles
- a person being compared to a group profile
- orthogonal vs correlated vector component profiles
- sten, standardized, or raw–score comparisons

The formula for comparing two individual profiles is:

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

$$r_p = \frac{E_k - \sum_{i=1}^{k} \left(z_{p_i} - z_{t_i} \right)^2}{E_k + \sum_{i=1}^{k} \left(z_{p_i} - z_{t_i} \right)^2}$$

where

z_{p_i} = the standard score of a comparison profile attribute i of k elements (scores)

e.g. $z_{p_i} = \frac{\left(p_i - \bar{p} \right)}{s_p}$ where

s_p = the standard deviation of the comparison profile scores

Likewise for the target profile scores.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

$E_k =$ the chance-expected sum of squared deviations between two profiles of k elements

E_k is defined by considering the variance of the difference between two variables ...

$\sigma^2_{p-t} = \sigma^2_p + \sigma^2_t$ .... and the expected value of a chi-square variable (the median value in this case)

Given a chi-square variable is the sum of squared values of a set of k standard scores, then the median value corresponds to the 50th percentile of a distribution of all such possible scores, for $df = k$.

This value multiplied by the sum of variances of the two sets of profile scores provides the value of E_k for any particular profile comparison that varies in terms of k attributes.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

So: for **standard scores** with an SD and variance of 1.0, the value of E_k is:

$$E_k = \chi^2_{50} \cdot \sigma^2_p + \sigma^2_t = \chi^2_{50} \cdot (1+1) = 2 \cdot \chi^2_{50}$$

Given the median χ^2 for any $\mathrm{k} \approx \mathrm{k}$, then this is why Cattell (1969, 1988) sometimes defines it as:

$$E_k = 2 \cdot k$$

whereas for **sten scores** (with an SD of 2.0, and a variance of 4), the value of E_k is:

$$E_k = \chi^2_{50} \cdot \sigma^2_p + \sigma^2_t = \chi^2_{50} \cdot (4+4) = 8 \cdot \chi^2_{50} = 8 \cdot k$$

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

The coefficient varies between +1.0 and -1.0, with +1.0 indicating maximum similarity, -1.0 = maximum inverse similarity, and 0.0 indicating no relationship whatsoever.

The coefficient is sensitive only to shape, as both elevation and scatter are equated by the deviation and normalisation by standard deviation

	r_p
Raw profile comparison	0.99
De-Elevation Comparison	0.99
De-Scatter Comparison	0.99

But…

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

This comes as no surprise as the data are being normalized and standardized prior to any calculation, which is equivalent to de-scattering the data (equating for elevation and variability). However, it does serve as a warning that using r_p via the standardized data formula removes sources of disagreement between the data that might otherwise be considered significant. The two test two profiles are different - they retain the same shape but there is a substantive level difference between them.

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

One simple way to avoid the issue is to compute the r_p using the actual data values themselves, prior to any transformation. Remember, to do this requires calculating the correct chance-expected squared deviation difference between two such profiles. From the above equations, using the variances for the two raw data profiles, we obtain for raw and de-scattered profiles respectively:

$$E_k = \chi^2_{50} \cdot \sigma^2_p + \sigma^2_t = \chi^2_{50} \cdot \left(32.447 + 7.538\right) = 39.985 \cdot \chi^2_{50}$$

$$E_k = \chi^2_{50} \cdot \sigma^2_p + \sigma^2_t = \chi^2_{50} \cdot \left(0.0909 + 0.0909\right) = 0.1818 \cdot \chi^2_{50}$$

Transformation Sensitivity: distance-based coefficients

Profile Similarity Coefficient

Standardized data coefficients

	r_p
Raw profile comparison	0.99
De-Elevation Comparison	0.99
De-Scatter Comparison	0.99

Raw data coefficients

	r_p
Raw profile comparison	-0.20
De-Elevation Comparison	0.64
De-Scatter Comparison	0.99

Look at the negative coefficient – somewhat "out of sync" with the actual form of the raw profiles …

Transformation Sensitivity: shaped-distance coefficients

shaped distance

The initials ADD stand for **Accelerated Distance Degredation**. The reason for using such coefficients is when a user wishes to increment the distance between two objects in space, at a rate or in steps that are related nonlinearly to the actual euclidean or signed difference. In actuality, there is no actual coefficient per se - but rather a wide variety of "distance" functions which provide a direct but "shaped-distance" measure.

Transformation Sensitivity: shaped-distance coefficients

shaped distance

*I have referred to this property of ADD coefficients as "**shaped distance**" as its use is in direct response to a need to "take control" or "shape" the distance function between two objects in space, as a function of their distance.* The GeneSys 3 profiling module achieves a form of ADD by permitting the weighting of scales used in a profile. This weighting can be linear, non-linear, or even categorical. The problem with this is that the behaviour of the coefficient (its ranges of possible values) is no longer known, as it is likely to reflect a unique combination of weight functions applied to a conventional similarity index (which admittedly is a problem with all weighting schemes).

Transformation Sensitivity: shaped-distance coefficients

shaped distance

A particular design specification was developed for the Mariner7 profiling application – where small disparities between two profile scores would have little effect on the distance calculation, but, as the disparity grew greater, so would the distance be degraded in an accelerated fashion. The function chosen to achieve this was the normal distribution curve equation, with *degradation acceleration controlled by the standard deviation (sd)* parameter and the *target profile value represented by the mean* of the distribution function.

Transformation Sensitivity: shaped-distance coefficients

shaped distance

The ADD coefficient range is scaled to assess similarity between profiles, in a (0 to 100 metric) with the range of possible values extending between 0% (complete dissimilarity) to 100% (identity). The equation is:

$$score = \left[\frac{1}{\sigma\sqrt{2\pi}} e^{-\left[\frac{(p-t)^2}{2\sigma^2}\right]} \right] \cdot \left(100 \cdot \left(\sigma \cdot \sqrt{2\pi}\right)\right)$$

where

$\sigma =$ standard devation unit

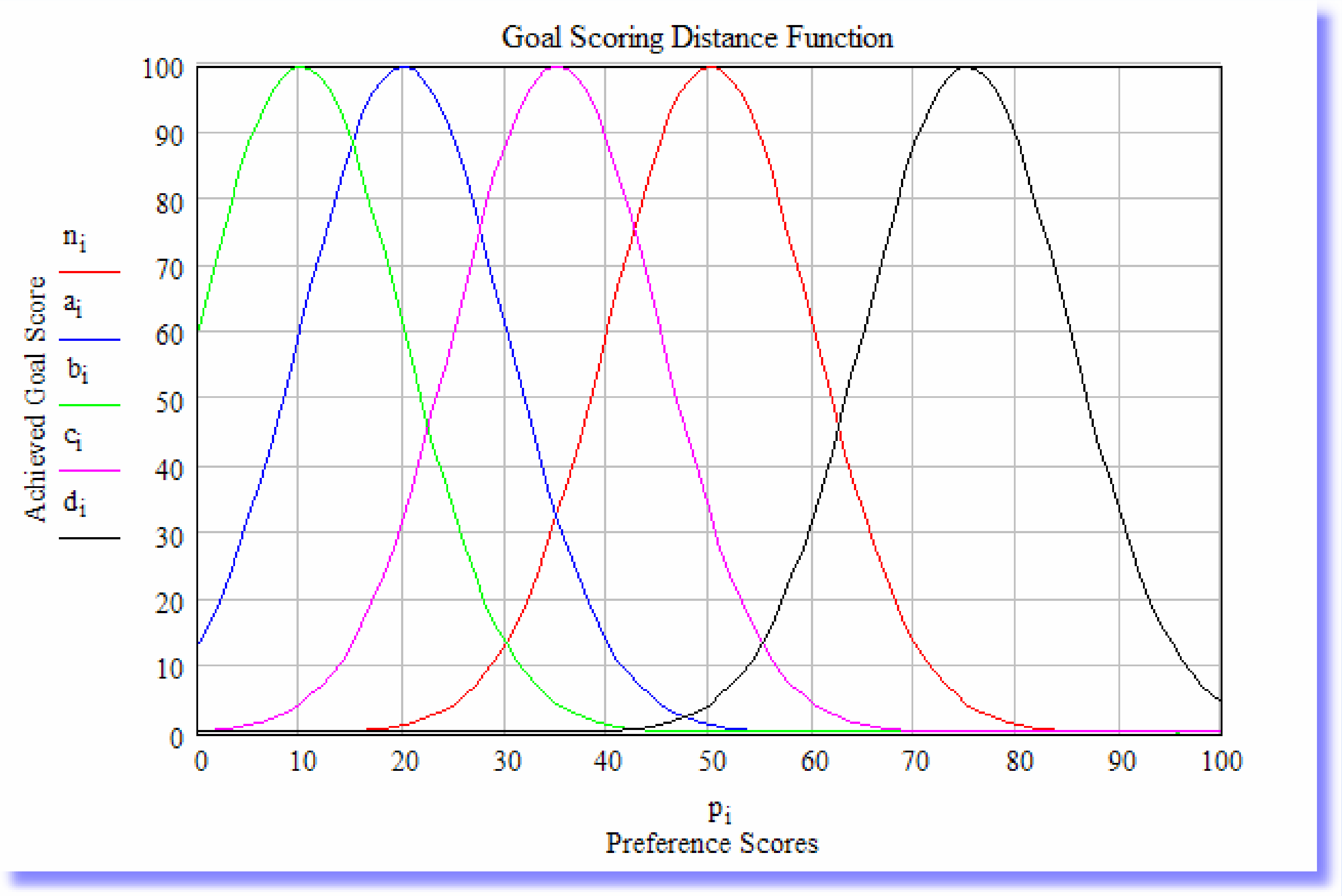
$p =$ person profile attribute value

$t =$ target profile attribute value

Transformation Sensitivity: shaped-distance coefficients

shaped distance

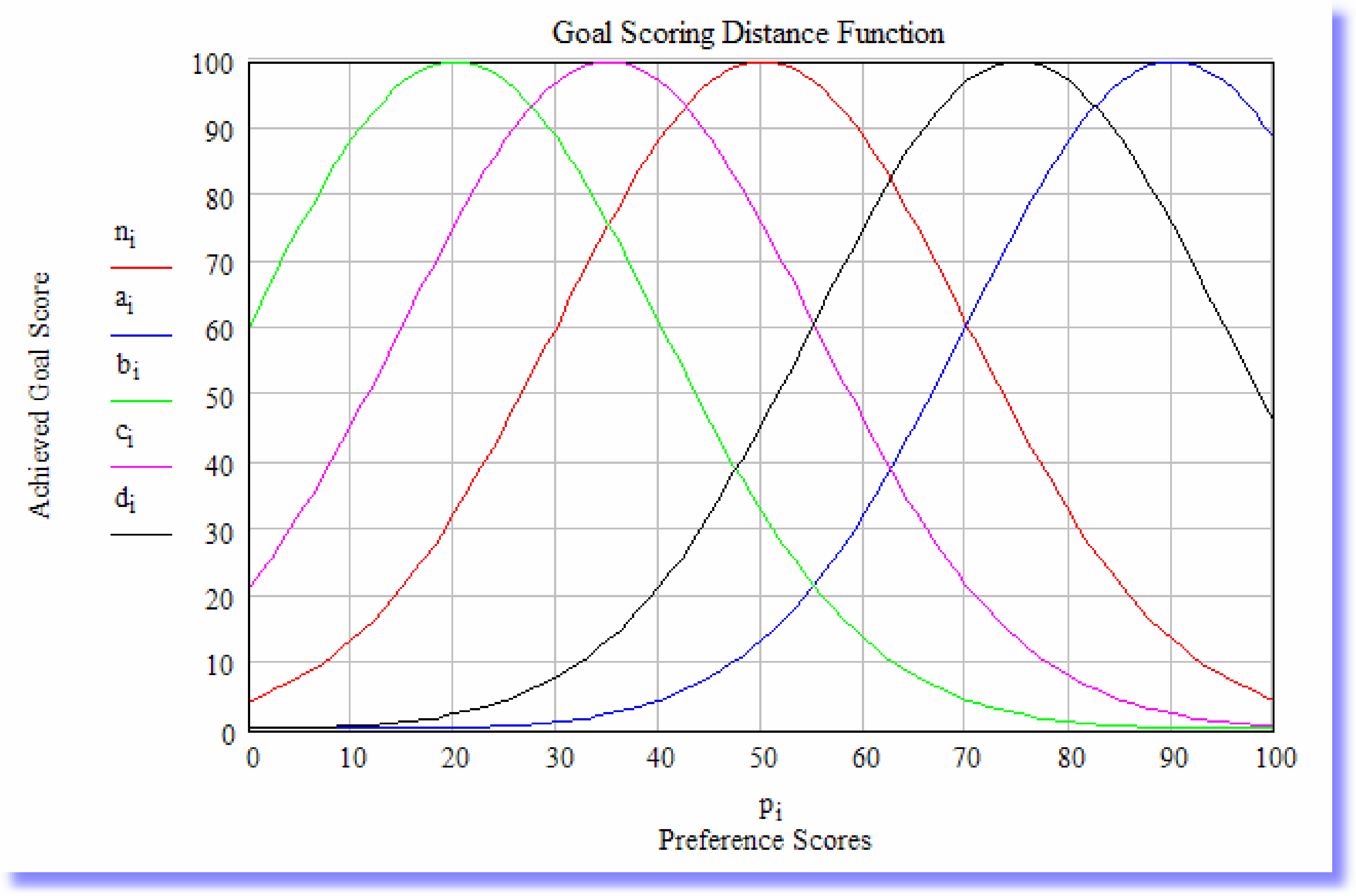
A standard deviation of 10



Transformation Sensitivity: shaped-distance coefficients

shaped distance

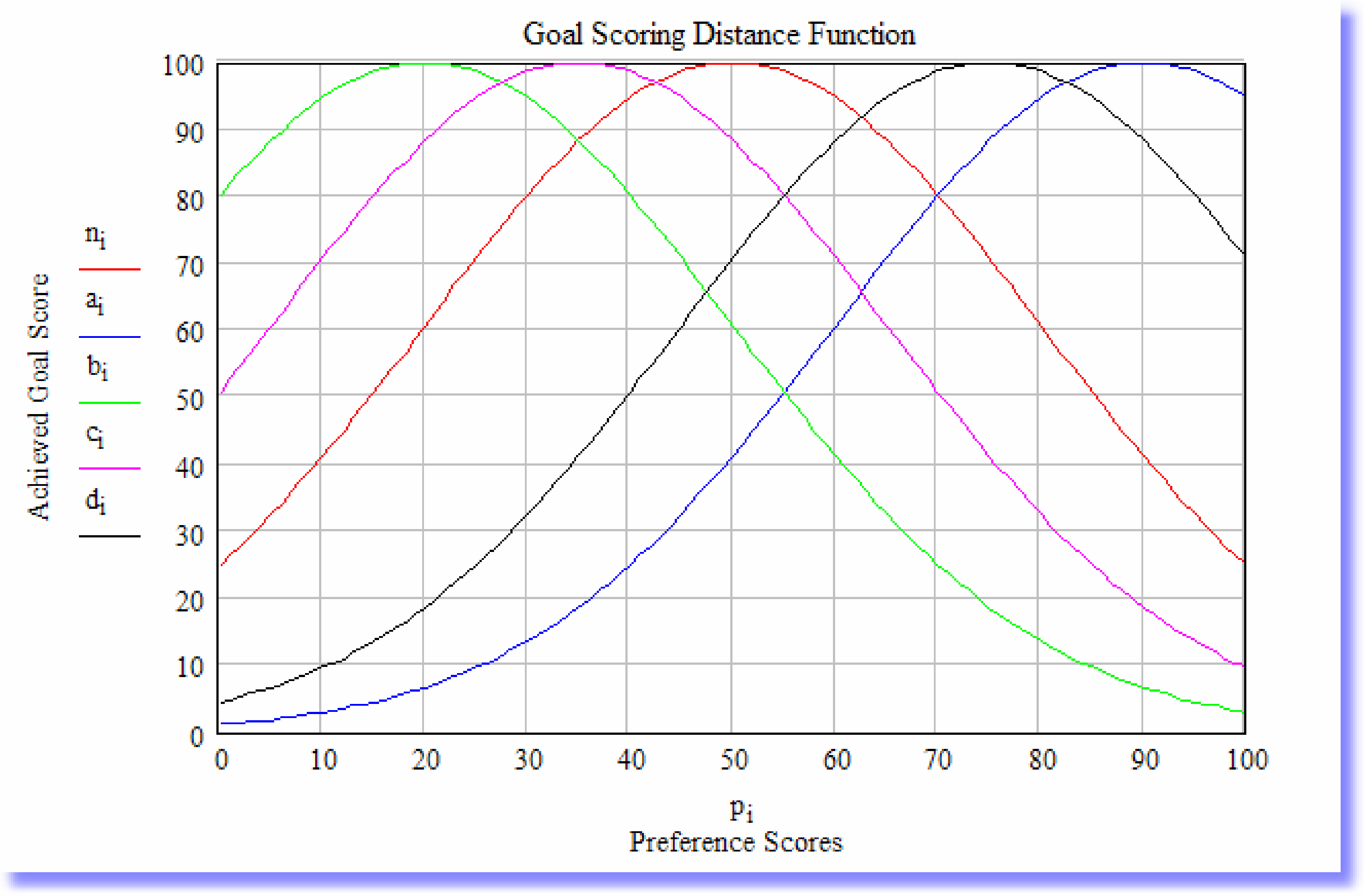
A standard deviation of 20



Transformation Sensitivity: shaped-distance coefficients

shaped distance

A standard deviation of 30



Transformation Sensitivity: shaped-distance coefficients

shaped distance

As I increase the standard deviation through 10 to 20, so I can enhance the "plateau" area whilst still maintaining accelerated degredation (over and above simple linear differencing). Note also, I am reporting distance in the same metric as my measurement.

In essence, what I do is compute the value of the point on the curve corresponding to the person profile attribute value, relative to a distribution with:
mean = target profile attribute score
sd = 20 (in the case of the Mariner7 profile scores)

This value is my computed "distance" which is expressed as a % similarity.

Transformation Sensitivity: shaped-distance coefficients

shaped distance

In designing the "shaped distance", there are no mathematically optimal values. Instead, this is a weighting designed to be empirically calibrated by a user so as to maximise the agreement between what looks similar and dissimilar by eye, and what the profile comparison index appears to be telling you.

Transformation Sensitivity: shaped-distance coefficients

shaped distance

The ADD coefficients for raw and transformed data using an **SD = 5.0** (slow degredation) are :

	ADD
Raw profile comparison	42.89%
De-Elevation Comparison	86.47%
De-Scatter Comparison	99.99%

The ADD coefficients for raw and transformed data using an **SD = 2.0** (faster degredation) are:

	ADD
Raw profile comparison	4.86%
De-Elevation Comparison	55.62%
De-Scatter Comparison	99.99%

The coefficient is differentially sensitive to elevation, scatter, and shape.

Transformation Sensitivity: shaped-distance coefficients

shaped distance

As an aside, the *shaped distance* concept is not really new – rather, it is a kind of analogue of what is referred to as kernel smoothing in exploratory data analysis, specifically in nonparametric data smoothing.

The kernel function is a probability function that is used to weight points in the computation of each local smoothing estimate. Points farther away from the estimate are usually weighted less than points nearer. The kernel function is in essence providing those weights. There are seven major kernel functions – the Normal/Gaussian is the one I use.

The Purpose

❷ To determine the expected size and frequency distribution of all possible values for each of the 9 profile coefficients, given a completely random dataset and a real-world personality test dataset (*distribution analysis*)

This is where the program "Coefficient Distribution Analyzer" is used. This program uses two datasets, one consisting of profiles of 10 normally distributed random numbers, randomising data via rows (cases), under the constraint that each attribute value had to be constrained between 0 and 60, whilst being normally distributed with mean 30 and standard deviation of 10.

Coefficient Distribution Analyzer

This is where the program "Coefficient Distribution Analyzer" is used. This program uses two datasets.

The first consists of 2011 profiles of 10 normally distributed random numbers (randomising data via rows (cases)) under the constraint that each attribute value had to be constrained between 0 and 60, whilst being normally distributed with mean 30 and standard deviation of 10.

The other consists of 2011 real-data cases of the Psytech International Occupational Personality Profiler test which consists of 10 personality constructs.

Coefficient Distribution Analyzer

With regard to the numbers of cases used for coefficient distribution analysis, I settled on **2011** (which yielded 2,021,055 profile coefficients for each dataset). That is, every case is uniquely compared to all other cases.

This is so as to obtain "hard" expected values for the various sizes of coefficients used on both random and real data. Also, this provides coefficient distributions that are virtual population distributions (such that you could report “the likelihood of observing a coefficient as large as 0.70 or greater by chance alone is ....". I have provided both graphics and coefficient histogram tables in the program for this purpose.

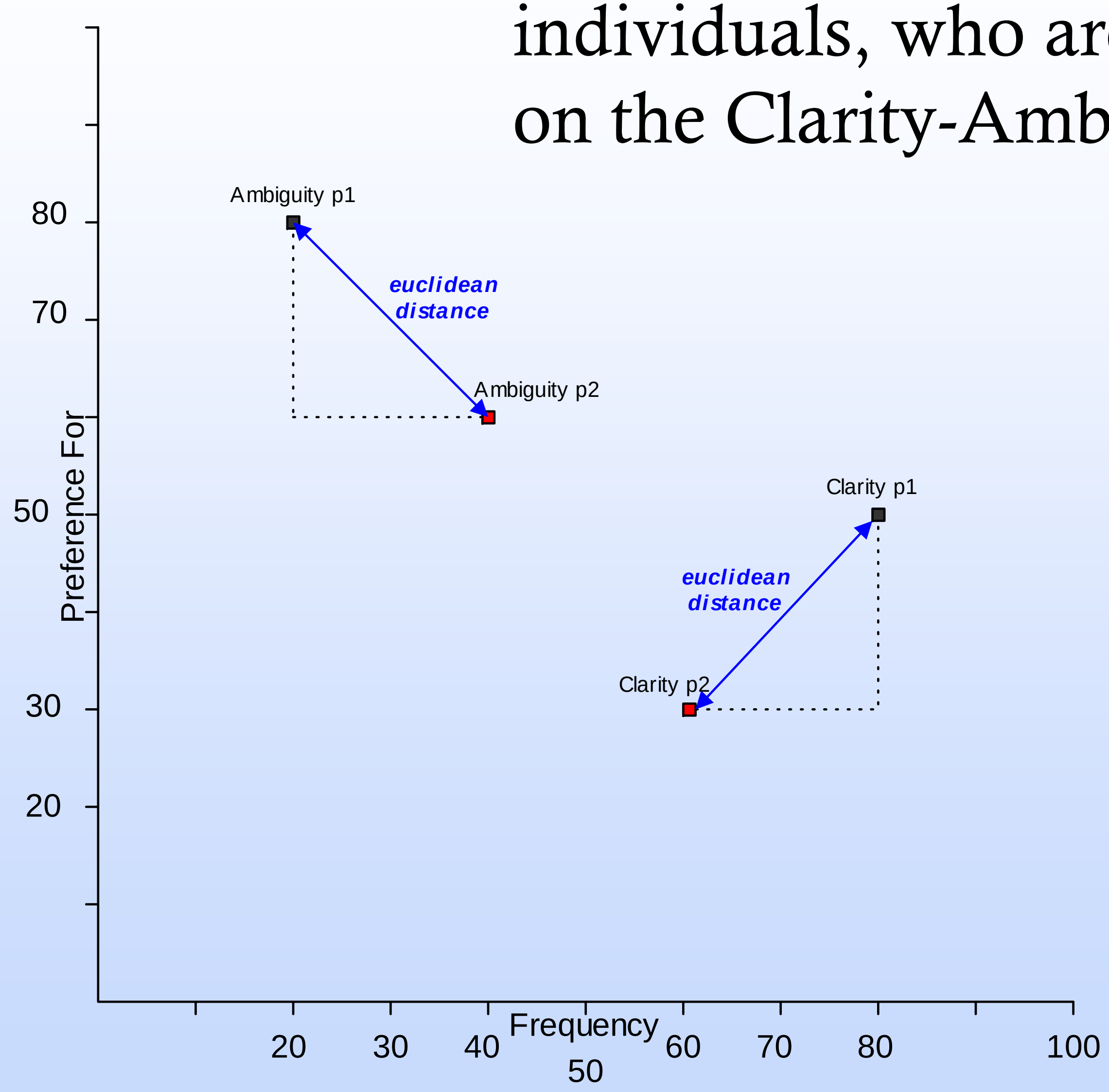
Coefficient Distribution Analyzer

2-D Order-Constrained Profile Similarity

A 2-dimensional shaped, accelerated distance degredation coefficient that not only takes into account the disparity between profile and target attributes, but also the order-relations between them.

2-D Order-Constrained Profile Similarity

The graph below shows two individuals, who are being compared on the Clarity-Ambiguity Triad.

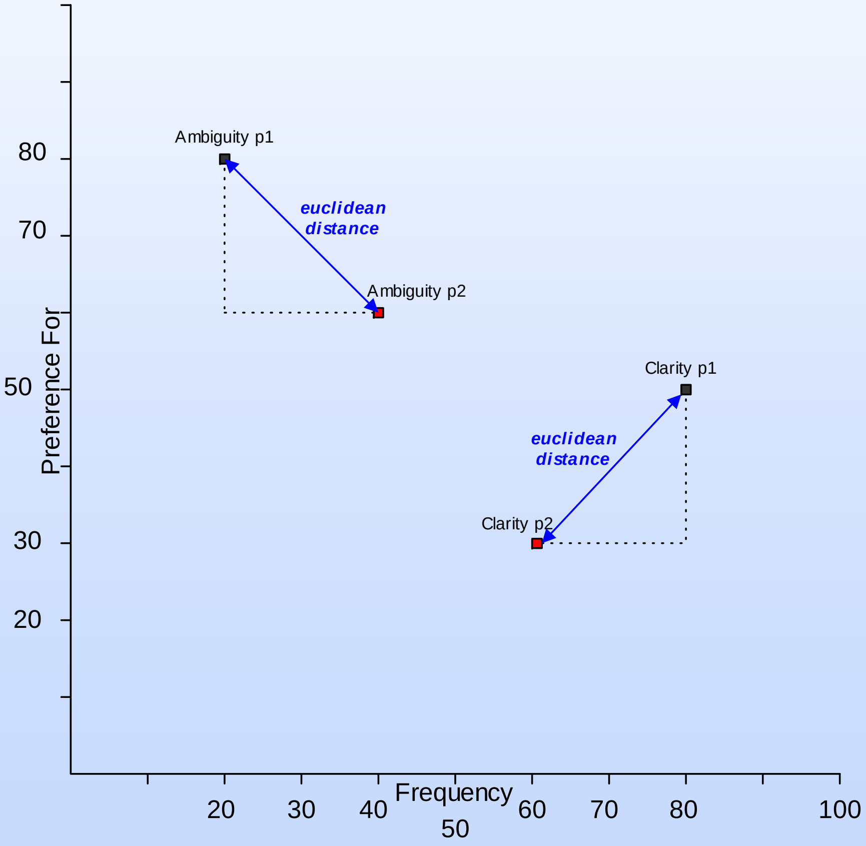


2-D Order-Constrained Profile Similarity

The original Mariner7 person-to-person match coefficient

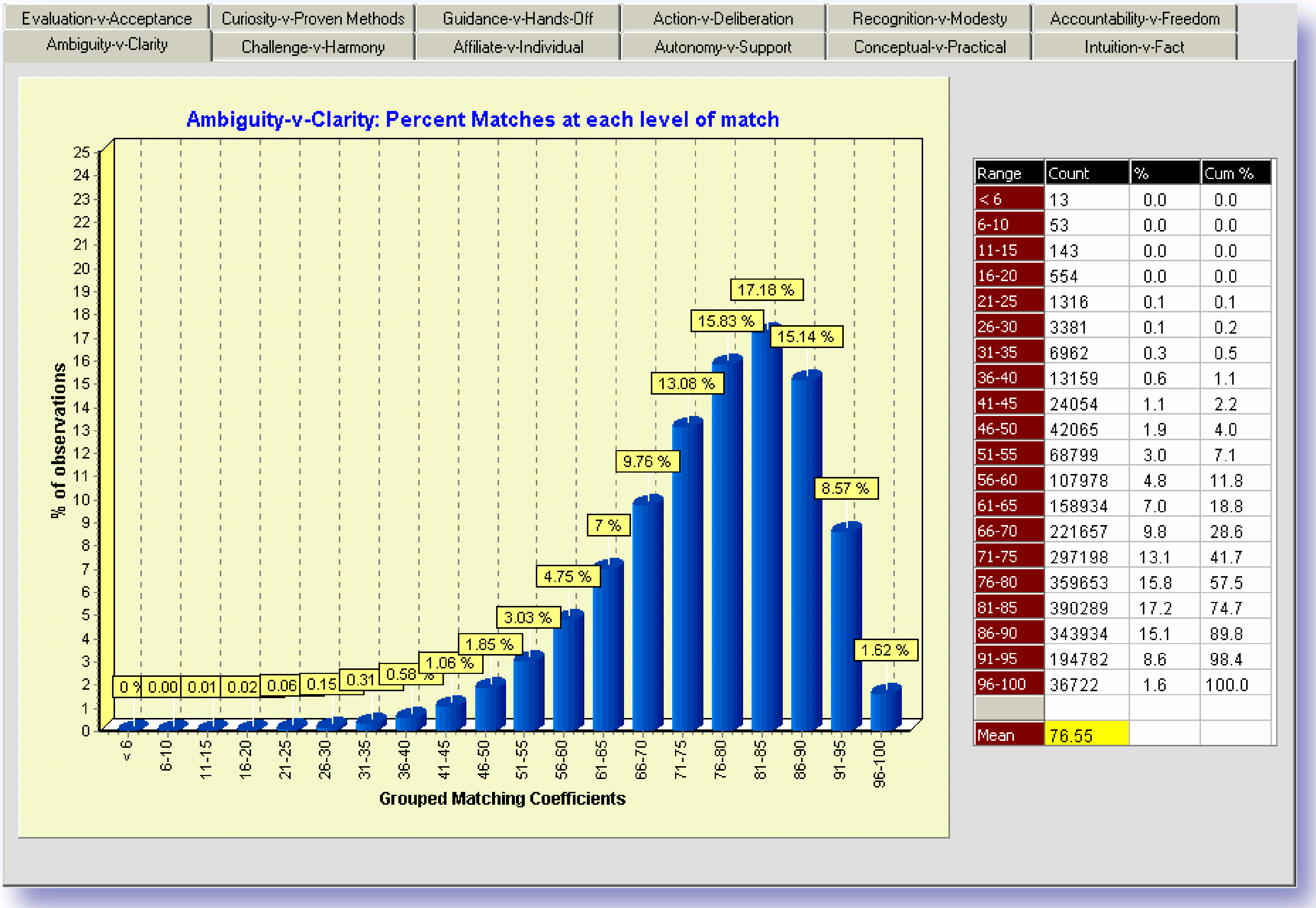
$$Match_{jki} = 100 - \left[\left(\left| a_{i1j} - a_{i1k} \right| \cdot 0.25 \right) + \left(\left| a_{i2j} - a_{i2k} \right| \cdot 0.25 \right) + \left(\left| f_{ij} - f_{ik} \right| \cdot 0.5 \right) \right]$$

$$Match_{jk1} = 100 - \left[\left(\left| 50 - 30 \right| \cdot 0.25 \right) + \left(\left| 80 - 60 \right| \cdot 0.25 \right) + \left(\left| 80 - 60 \right| \cdot 0.5 \right) \right] = 80\%$$



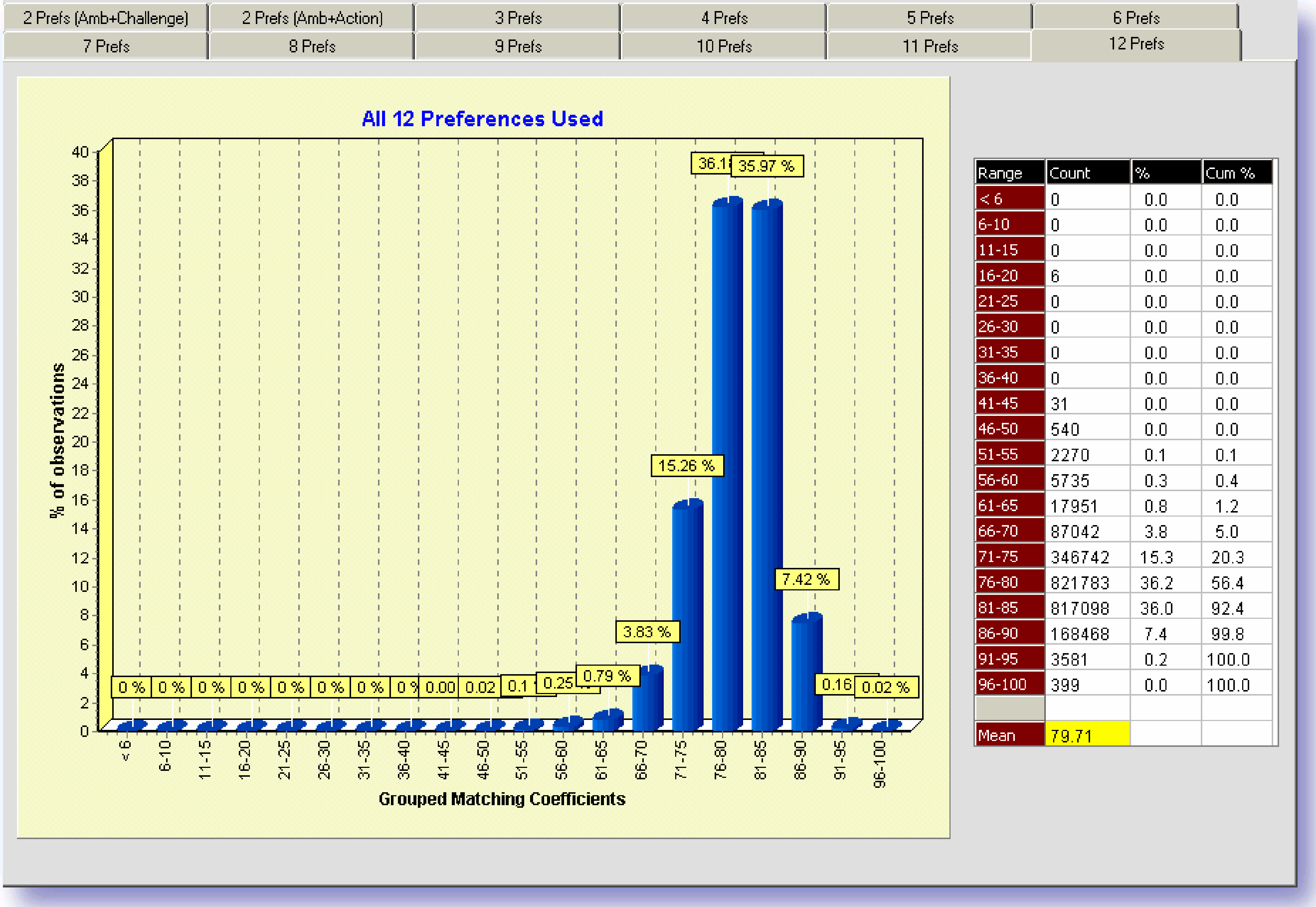
Using 2131 real-data cases, I compared every individual’s profile with every other individual (2.2million match coefficients) – and looked at the expected-value frequency distributions …

2-D Order-Constrained Profile Similarity



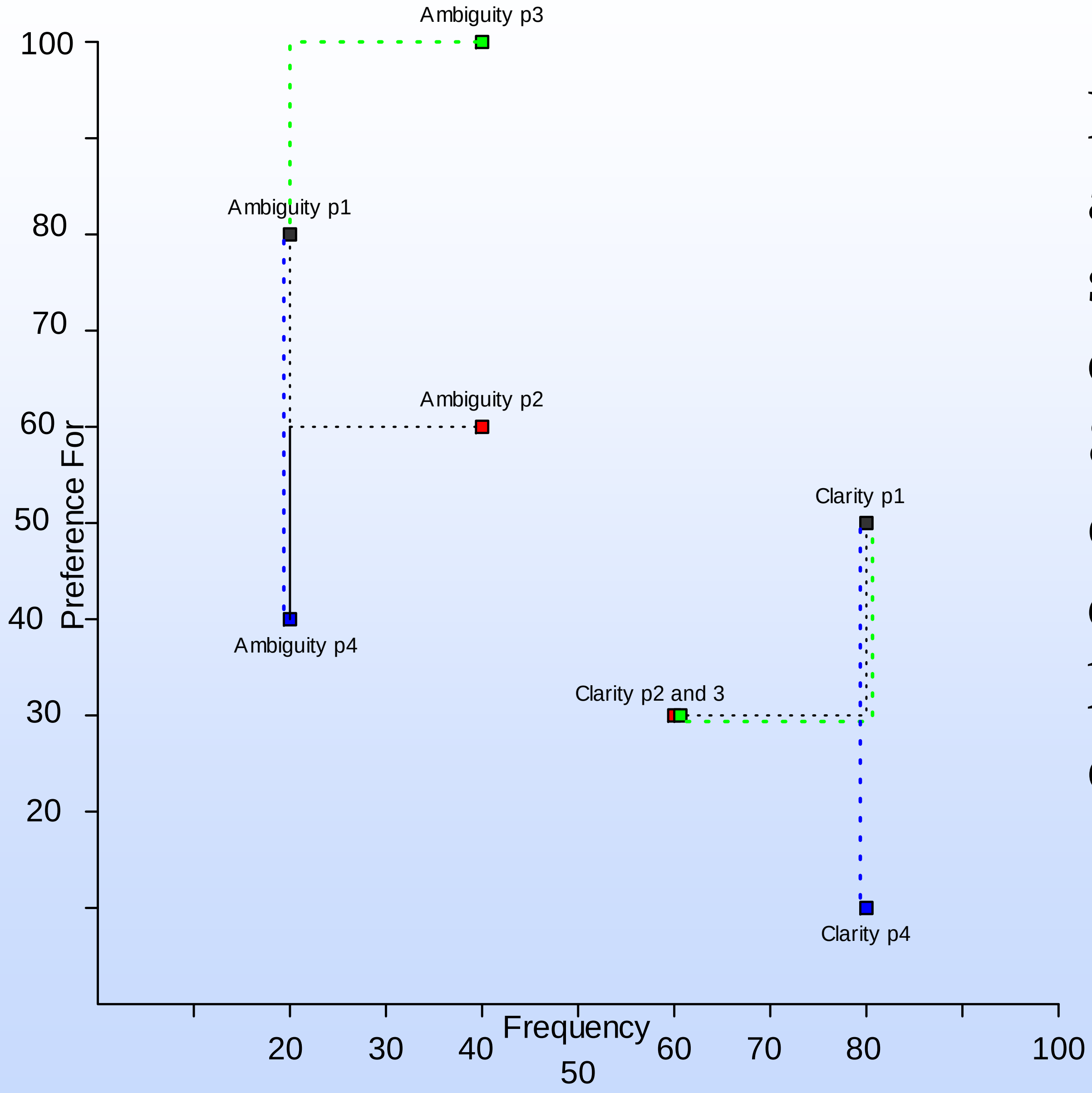
Range	Count	%	Cum %
< 6	13	0.0	0.0
6-10	53	0.0	0.0
11-15	143	0.0	0.0
16-20	554	0.0	0.0
21-25	1316	0.1	0.1
26-30	3381	0.1	0.2
31-35	6962	0.3	0.5
36-40	13159	0.6	1.1
41-45	24054	1.1	2.2
46-50	42065	1.9	4.0
51-55	68799	3.0	7.1
56-60	107978	4.8	11.8
61-65	158934	7.0	18.8
66-70	221657	9.8	28.6
71-75	297198	13.1	41.7
76-80	359653	15.8	57.5
81-85	390289	17.2	74.7
86-90	343934	15.1	89.8
91-95	194782	8.6	98.4
96-100	36722	1.6	100.0
Mean	76.55		

2-D Order-Constrained Profile Similarity



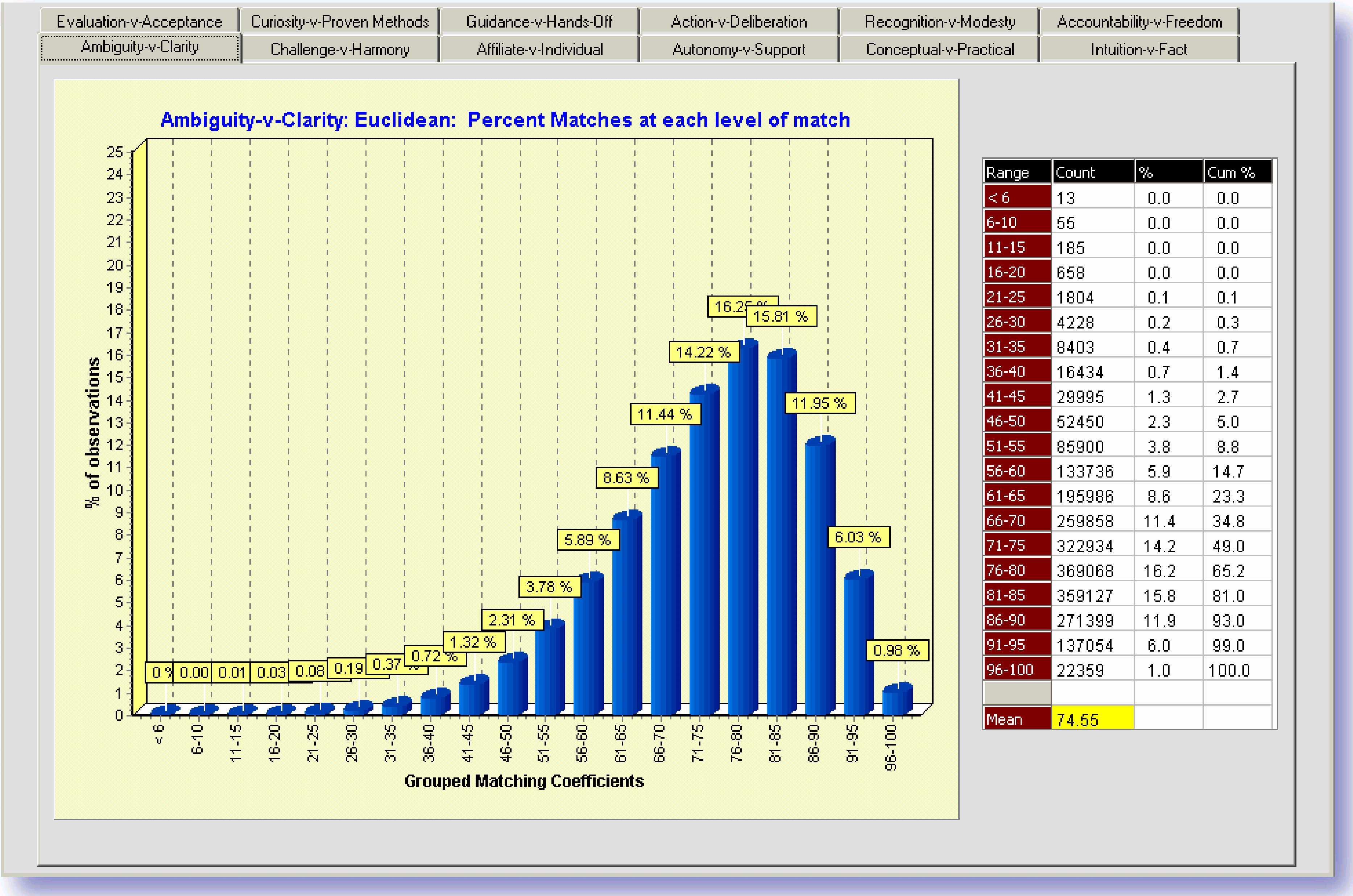
Range	Count	%	Cum %
< 6	0	0.0	0.0
6-10	0	0.0	0.0
11-15	0	0.0	0.0
16-20	6	0.0	0.0
21-25	0	0.0	0.0
26-30	0	0.0	0.0
31-35	0	0.0	0.0
36-40	0	0.0	0.0
41-45	31	0.0	0.0
46-50	540	0.0	0.0
51-55	2270	0.1	0.1
56-60	5735	0.3	0.4
61-65	17951	0.8	1.2
66-70	87042	3.8	5.0
71-75	346742	15.3	20.3
76-80	821783	36.2	56.4
81-85	817098	36.0	92.4
86-90	168468	7.4	99.8
91-95	3581	0.2	100.0
96-100	399	0.0	100.0
Mean	79.71		

2-D Order-Constrained Profile Similarity



Persons p1, p2, and p3 all have the same match M7 coefficient value of 80%. Euclidean distance coefficient calculations make little differentiation either …

2-D Order-Constrained Profile Similarity



Range	Count	%	Cum %
< 6	13	0.0	0.0
6-10	55	0.0	0.0
11-15	185	0.0	0.0
16-20	658	0.0	0.0
21-25	1804	0.1	0.1
26-30	4228	0.2	0.3
31-35	8403	0.4	0.7
36-40	16434	0.7	1.4
41-45	29995	1.3	2.7
46-50	52450	2.3	5.0
51-55	85900	3.8	8.8
56-60	133736	5.9	14.7
61-65	195986	8.6	23.3
66-70	259858	11.4	34.8
71-75	322934	14.2	49.0
76-80	369068	16.2	65.2
81-85	359127	15.8	81.0
86-90	271399	11.9	93.0
91-95	137054	6.0	99.0
96-100	22359	1.0	100.0
Mean	74.55		

2-D Order-Constrained Profile Similarity

In order to better represent similarity, an algorithm was devised to operate at the “triad” level:
a doubly degraded Euclidean distance using custom degredation functions based upon whether the data for each person have equivalent preference and frequency assignments in terms of difference between attribute 1 and attribute 2 within each person’s profile. In essence, we are ***accelerating*** the distance between two individuals attribute scores as they depart from one another, and also as they diverge from an order constraint on both the preference and frequency axis.

2-D Order-Constrained Profile Similarity

Step 1

- ✪ Compute the standard Euclidean Distance between the two attributes of a triad – computing the distance in 2-dimensional space between the two attribute scores for two individuals (e.g. compute the distance between person1-attribute1 and person2-attribute 1, and likewise for attribute 2).
- ✪ Express each raw Euclidean distance as a normalised value, by dividing through by the maximum distance possible given the 0-100 distance range on each axis.
- ✪ Subtract this normalised distance from 1.0 to express it as a similarity measure (concordant with how we express a match (in terms of similarity).
- ✪ Then, take the average of these two normalised similarities as the overall triad similarity between the two individuals, and multiply it by 100 to express it as a % match.

2-D Order-Constrained Profile Similarity

Step 2

- Compute the distance parameters **k1** and k2 against the **Preference** axis for the points.
- Also note whether Attribute A1 preference is higher, lower, or equal to Attribute A2 preference for Person 1.
- Now determine whether the same "order" constraint is present for Person 2.
- If the order constraint is the same for both persons, degrade the overall triad similarity using a custom formula.
- If the order constraint is different, then doubly degrade the similarity using an additional degredation function.

2-D Order-Constrained Profile Similarity

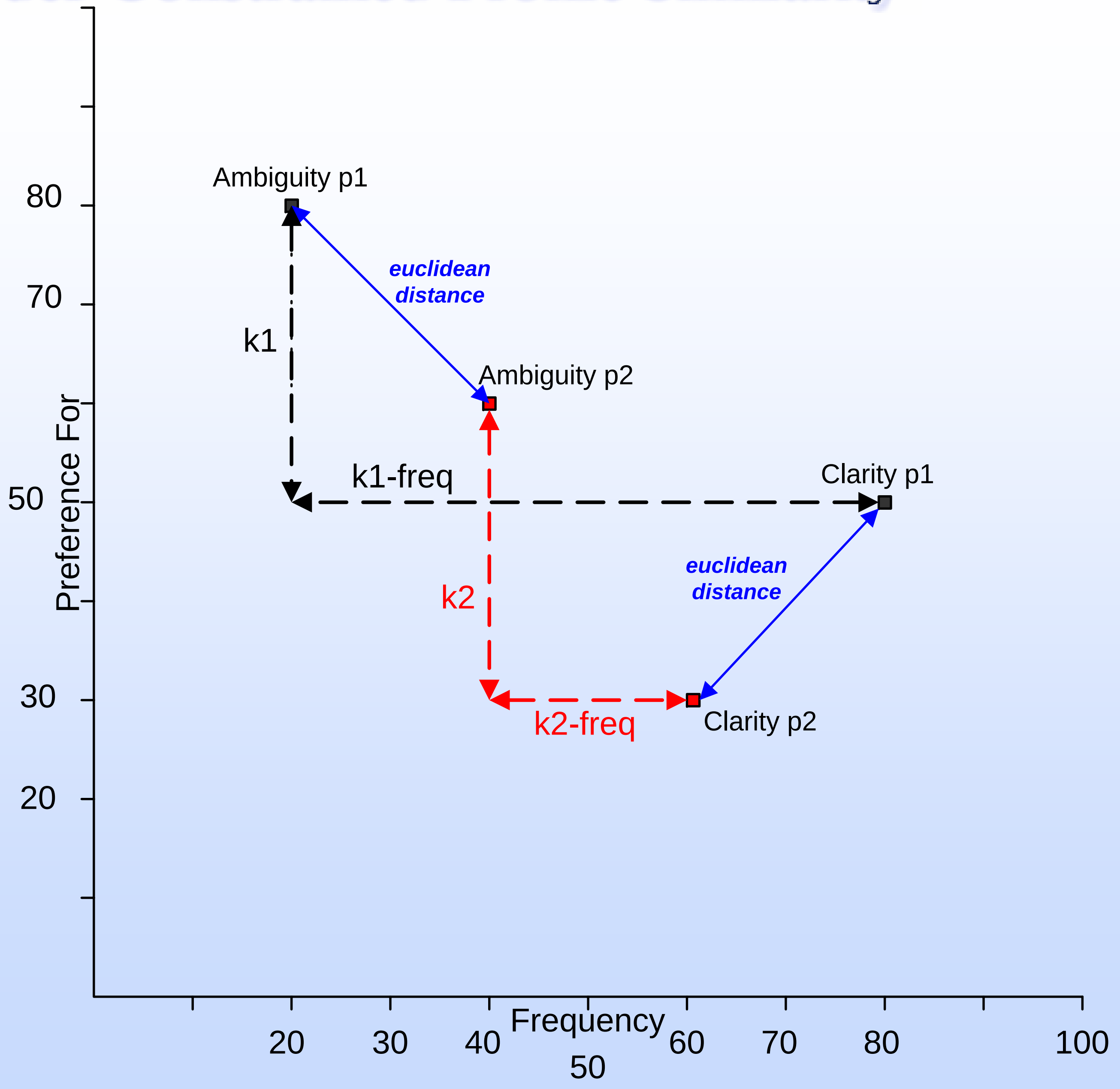
Step 3

- Compute the distance parameters **k1-freq** and **k2-freq** against the **Frequency** axis for the points
- Also note whether Attribute A1 frequency is higher, lower, or equal to Attribute A2 frequency for Person 1.
- Now determine whether the same "order" constraint is present for Person 2.
- If the order constraint is the same for both persons, degrade the overall triad similarity using a custom formula.
- If the order constraint is different, then doubly degrade the similarity using an additional degredation function.

2-D Order-Constrained Profile Similarity

In essence, *we compute a similarity value (match value), then degrade it according to the ratio of the difference* conditions on both the preference and frequency axes, in turn,

2-D Order-Constrained Profile Similarity



2-D Order-Constrained Profile Similarity

2-D Order-Constrained Profile Similarity

The document entitled:
“2-Dimensional Order-Constrained Profiling.doc”

Details the algorithms and formulae with complete worked examples and ready-made Delphi program code to implement the entire sequence of operations

Profile Degredation Analysis

The aim of these tests is examine coefficient behaviour by using sten profiles that visually look very similar, but that:

❶ differ dramatically in just one particular attribute (scale), with all other scale scores exactly equal.
❷ differ in terms of a fixed sten difference (1 sten and 2 sten difference for each profile attribute).

Profile Degredation Analysis

I have varied the number of profile elements/scale scores from 6 through 20. I have also reported the Genesys coefficient in comparison to 4 other coefficients:

❶ GeneSys r_p

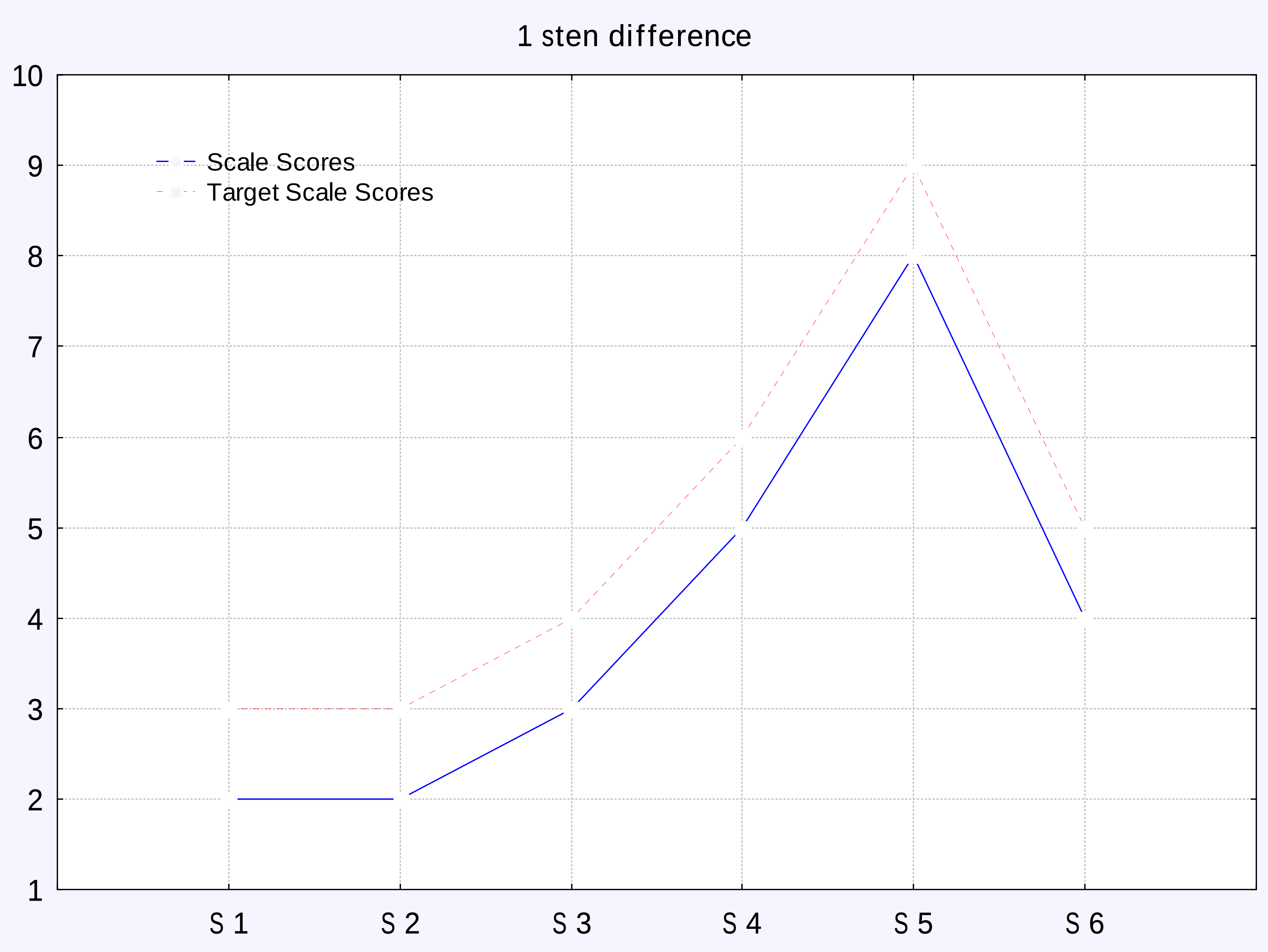
❷ Pearson r

❸ 1-Guttman's alienation

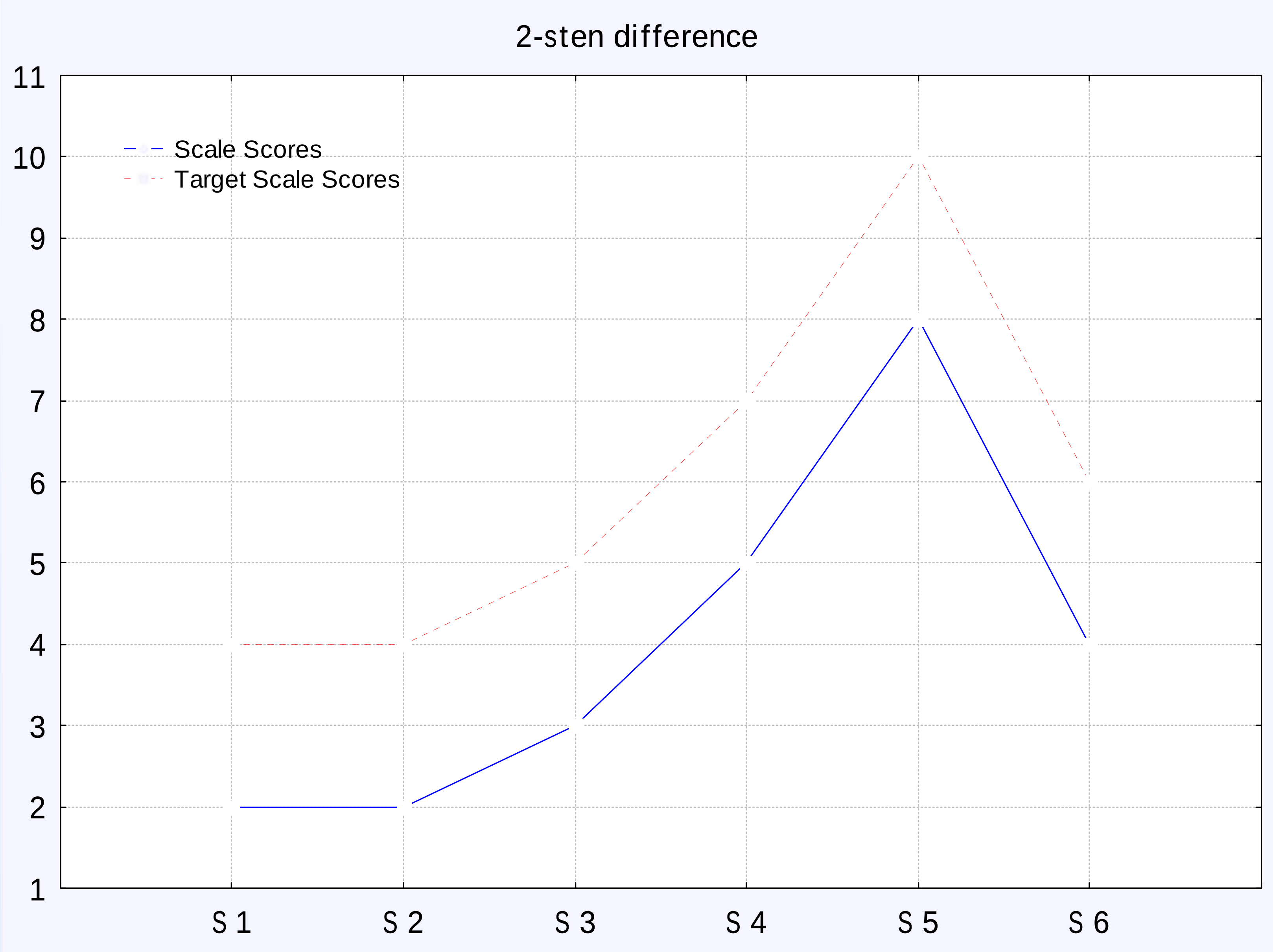
❹ Intraclass Model 2

❺ Barrett ADD (using a 1.6667 ratio SD parameter – 1/6th of 10 = SD of 1.6667)

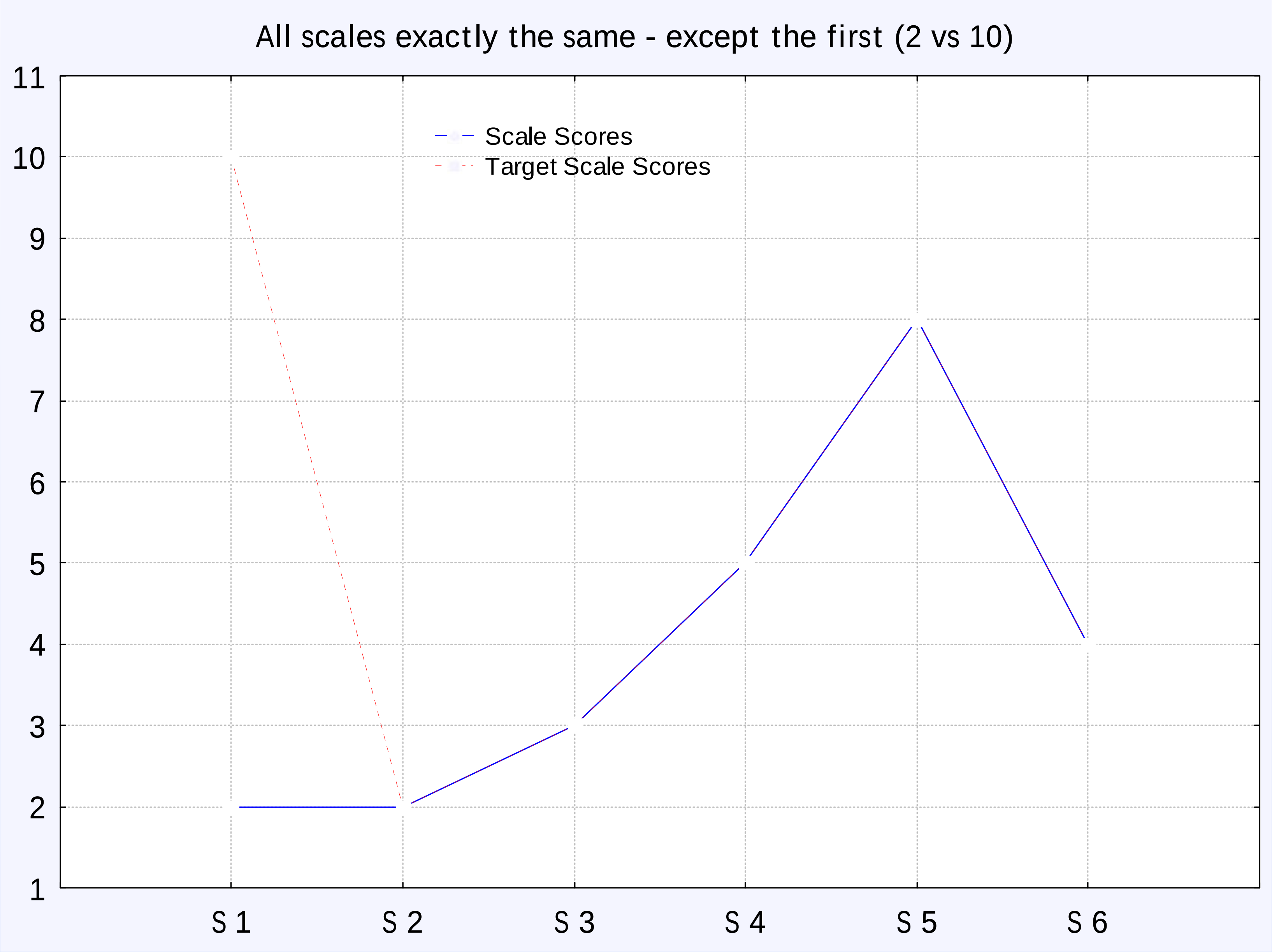
Profile Degredation Analysis



Profile Degredation Analysis

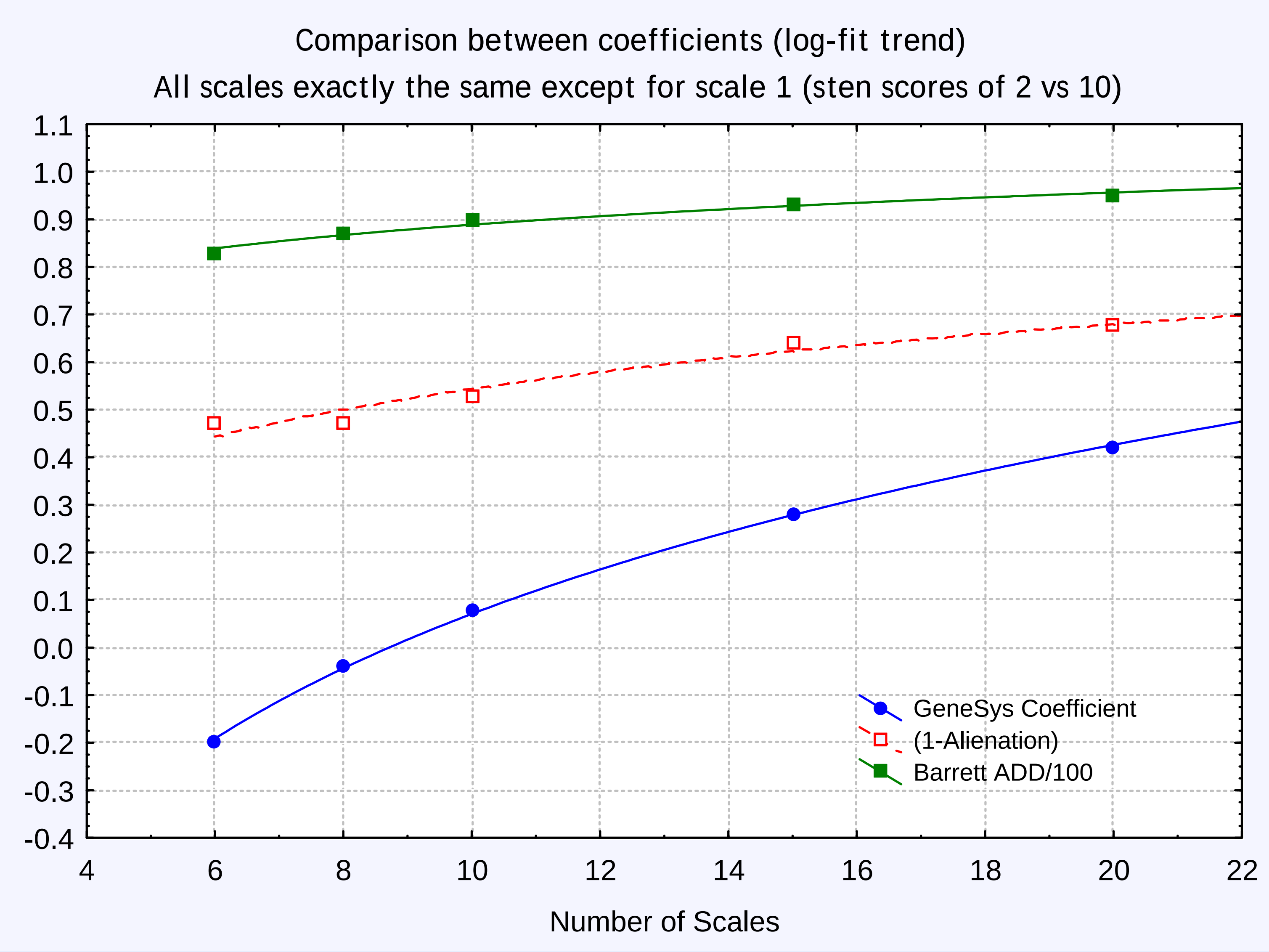


Profile Degredation Analysis



Profile Degredation Analysis

Graphing the GeneSys coefficient r_p coefficient size as a function of the number of **sten** scale scores, **given all scale scores are exactly equal, except for the first**, which is always a sten of 2 vs 10.



Profile Degredation Analysis

The complete analysis is in the document on the CD entitled:

GeneSys Profiling Coefficient Analysis #2.doc

** With kind permission from Psytech International Ltd. (for whom this work was undertaken).*